Victorian 6502 User Group Newsletter

KAOS

For People Who Have Got Smart

HARDWARE .. .. DAVID ANEAR
SOFTWARE .. .. JEFF RAE
FORTH .. .. DAVID WILSON
AMATEUR RADIO ROD DRYSDALE VK3BYU
EDUCATION .. .. JEFF KERRY
LIBRARY.. .. .. RON KERRY
TAPE LIBRARY . .. . JOHN WHITEHEAD
DISK LIBRARY WARREN SCHAECHE (B.H.)
NEWSLETTER .. .. IAN EYLES
SYM. .. ..BRIAN CAMPBELL
SECRETARY .. .. . ROSEMARY EYLES

OSI SYM KIM AIM UK101 RABBLE 65

Registered by Australia Post
Publication No. VBG4212

ADDRESS ALL CORRESPONDENCE TO 10 FORBES ST., ESSENDON, VIC. 3040

Vol.4 No.5 February 1984

Remember to bring all your old (unwanted) gear and/or money along to the sidewalk sale at the February meeting. The school will be opened at 1pm and the goods can be sold from your car, truck or whatever or in the main downstairs foyer. Please remember that everything you bring must be taken away again, either by you or the purchaser. We expect this to be a really bumper sale judging by what members have told us, even Compsoft is bringing lots of goodies to sell cheap.

Unfortunately Adam Dickson's listing of Floating Point FORTH could not be included in the newsletter, as even when we had it reduced until it was just legible, it was still too long to publish. We will have copies of the listing available at the meeting for anyone that's interested. If you won't be at the meeting and would like a copy just let us know and we will send you the listing post free.

In last month's newsletter John Whitehead asked for those members who had hardware or software for sale to contact him with details, so that a directory could be made up to assist other members in obtaining items of interest, so if you have something to sell please contact him now. Refer to page 15 of the January KAOS for the details he requires.

The next meeting will be on Sunday 26th February at 2pm at the Essendon Primary School which is on the corner of Raleigh and Nicholson Streets, Essendon. Please note that the school will be open at 1pm.

The closing date for articles for the March newsletter will be Friday 9th March.

INDEX

THE BEGINNING MACHINE LANGUAGE PROGRAMMER.....part 20
by David Dodds

BMLP last month dealt with ROL (ROtate Left) and ASL (Arithmetic Shift Left). This month we will put the merry-go-round in reverse gear and take a look at their backward cousins Rotate Right (ROR) and Logic Shift Right (LSR).

ROR
Study the diagram in fig.1 carefully in relation to the one for ROL last month.

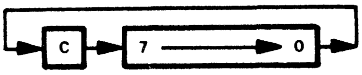


fig.1

The diagram shows that each bit involved in the ROR is moved one place to the right from bit 7 (the most signifigant bit) towards bit 0. This in turn is shifted to the carry bit in the status register and the former contents of carry is shifted to bit 7.

```
          eg   A=1   00000001   C=1
     after ROR A  (accumulator mode)
               A=128 10000000   C=1
     after another ROL A
               A=192 11000000   C=0
```

The ROR is therefore a backwards ROL. (There are those of course who would argue the reverse; that a ROL is a backwards ROR). Like the ROL the ROR is a closed loop system; none of the bits are lost or altered, only displaced.

Since the carry is involved directly in the ROR it must vary according to the value moved into it from bit 0. As well as this, both the negative (N) and zero (Z) flags are updated during the operation.

LSR
Logic shift right is a sort of mirror image of the ASL (almost). See fig.2 below.

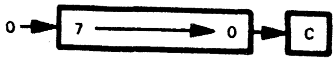


fig. 2

The direction of movement of bits is reversed being from bit 7 towards bit 0. Unlike the ASL there is no connection between bit 7 and carry. In the LSR bit 0 moves into the carry and a zero (0) is moved into bit 7.

```
          eg   A=$85 10000101   C=0
     after LSR A
               A=$42 01000010   C=1
     after another LSR A
               A=$21 00100001   C=0
```

Essentially a LSR is a simple binary divide by 2 in which the remainder if any is in the carry.
ROR and LSR are used together to form multiple precision division routines.

Take a look now back to the HEX to BINARY routine in BMLP 19. Now write a routine using LSR and ROR to convert a value in location $00 from BINARY to HEX.

You can use the $BF2D screen driver in the BASIC ROM to display the result. Jumping back to the monitor at $FE2A will avoid clearing the screen so that you see the result of your efforts.

To finish up this month an instruction which does nothing:- NOP No OPeration.
A NOP does not alter the status registers in any way, all it does is kill time.

NOP is an inherent mode instruction. While at first NOP may appear quite useless it is an ideal way of patching out short sections of unwanted code. It is also handy for introducing measured delays into routines without upsetting anything.

NEWS FROM COMPSOFT

Compsoft is under new management with George's brother, Minos taking charge. We have been told that Compsoft will still endeavour to support OSI and Rabble 65 users and the person to talk to with any queries etc. is Paul Dodd.

For those people who are patiently waiting, the good news is that CP/M is up and running and will be demonstrated at the February meeting. The approximate price for the software and the two P.C. boards (Z80 board and disk controller board), completely built is $390.

This month's super special from Compsoft is the CP-80 (or equivelant) printer for $375 inc. tax

Compsoft will have lots of goodies at the sidewalk sale at the February meeting, Tasan video boards (bare) $20, Rabble Expansion boards $50, 2x Superboards price neg. and lots of other bits and pieces.

MEETING-SYDNEY

by Norman Bate

The December meeting of the SYDNEY OSI group took place as usual at Lugarno. There were 13 members with 6 computers and 1 visitor.

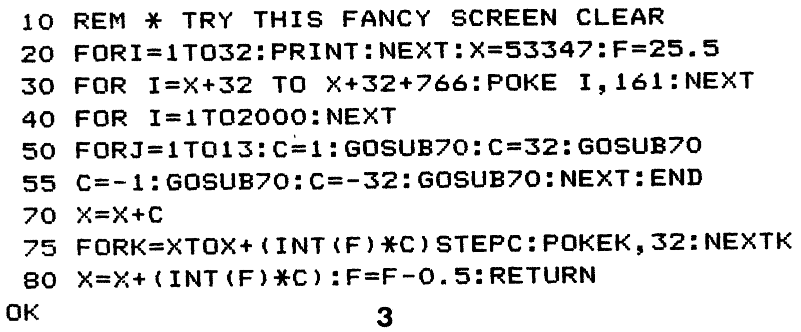
Visitor Tony Brown demonstrated his highly modified C1P MF running CP/M 2.2 with WORDSTAR etc. The CP/M mod consists of a vero board with a Z80 and Monitor ROM plus a few support chips. The 6502 and its Monitor ROM are removed from the C1P and the vero board plugged into the 6502 socket. The rest is all done by software.

Tony has also rewritten OS-65D V3.3 DOS to allow continuous addressing of both sides of a double sided drive. That is, he now has a directory using tracks 01-78. Details are available if required.

The next meeting is at Lugarno Girl Guides Hall on Sunday February 26th at 10am. For information contact N. Bate or N. Bissett

FANCY SCREEN CLEAR FOR 24x24

by Gerard Campbell

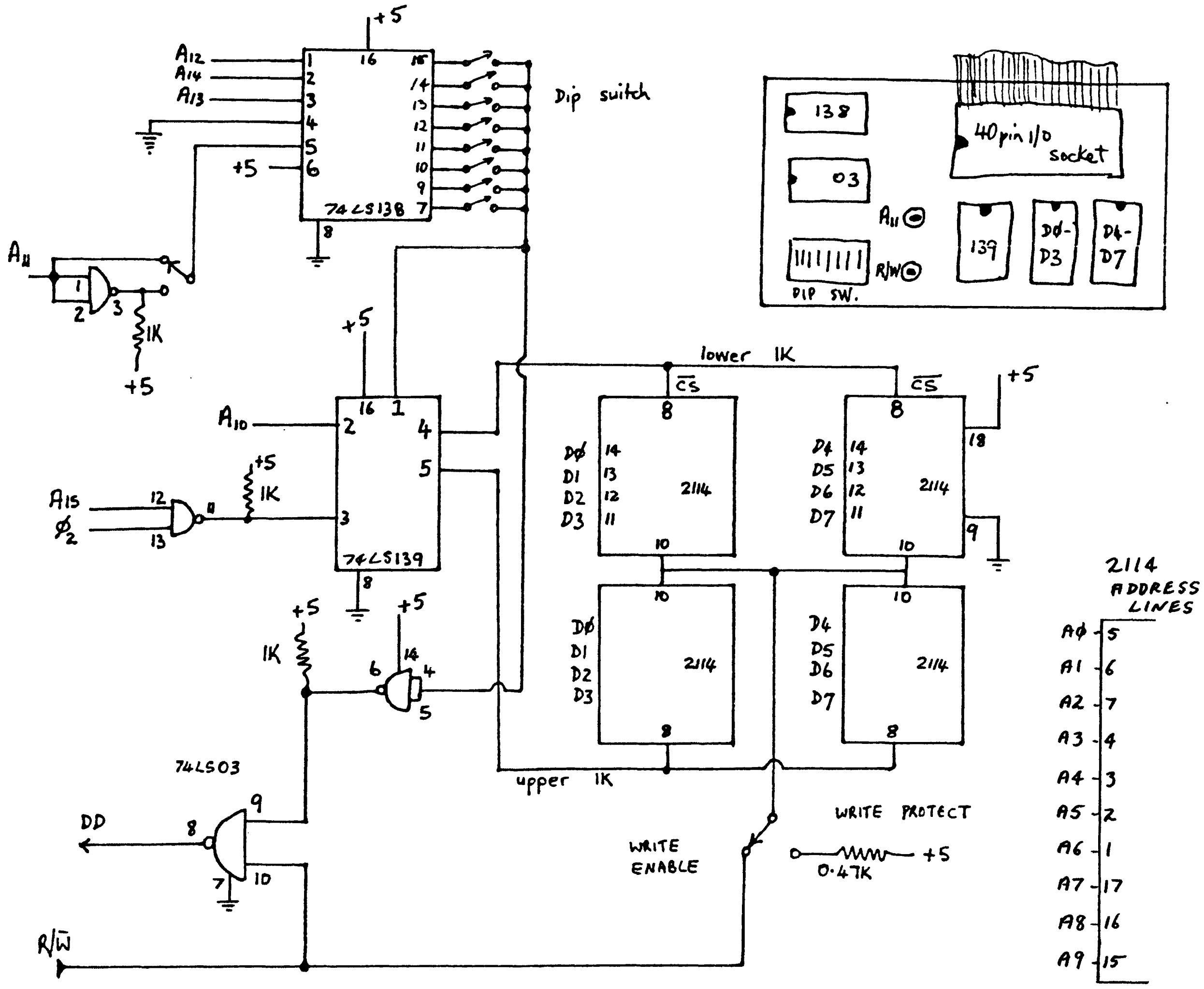


```
10 REM * TRY THIS FANCY SCREEN CLEAR
20 FORI=1TO32:PRINT:NEXT:X=53347:F=25.5
30 FOR I=X+32 TO X+32+766:POKE I,161:NEXT
40 FOR I=1TO2000:NEXT
50 FORJ=1TO13:C=1:GOSUB70:C=32:GOSUB70
55 C=-1:GOSUB70:C=-32:GOSUB70:NEXT:END
70 X=X+C
75 FORK=XTOX+(INT(F)*C)STEPC:POKEK,32:NEXTK
80 X=X+(INT(F)*C):F=F-0.5:RETURN
OK
```

Superboard

February, 1984.
NEWSLETTER OF THE OHIO SUPERBOARD USER GROUP, 146 YORK STREET, NUNDAH, 4012.

ADD SCRATCHPAD RAM TO YOUR SUPERBOARD by Bernie Wills.



By building up this small circuit, possibly with parts from the junkbox, you can have 2K of scratchpad ram at any address from $8ØØØ up to $F8ØØ.

The inset (top right) in the drawing, shows the layout I used. This shows only 2 x 2114 RAM chips, as the other 2 are piggybacked on them.

The Write Enable/Protect allows you to use the scratchpad ram as temporary rom, while the All switch in combination with the operation of one of the Dip switches enables the selection of any 2K within the range above.

If you have any queries or problems with the circuit, send a SAE to Bernie Wills,

Next Month: Word Puzzle Solver, NOS Basicode for OSI, Your Financial Whiz., Review of Night Rider game.

Coming: Adventuring part two, Simultaneous Equation solver, Soldering - some tips to make you a Pro., Boolean operators.

— SUPERBOARD —

This case study may interest readers, particularly those developing marketable items.

ROOKY I SEE

For about 18 months I have been selling replacement character sets for OSI computers. On 1.7.83, I received from Bruce Raymond, 2 Herbert St., Watsonia, a letter requesting three 2716 character sets - Enhanced, Scientific and Higher Resolution. He also wanted two 2732's for a friend who dropped in. The goods were forwarded on 4.7.83, with the usual notes and a letter showing how any number of 2716's could be switched.

At the recent OSUG meeting in Brisbane, I was more than a little surprised to read a Looky Video catalogue dated July 1983. It showed a photocopy of material I send to those buying or enquiring about character sets, and a price list showing Enhanced, Scientific and Higher Res. graphics in 2716's or 2732's, the irresistible margin of $0.80 less than my usual prices. No authorisation for this advertising or selling was sought by, nor given to, Looky Video.

I have an old catalogue from that firm listing the name 'Bruce Fisher'. The Melbourne telephone directory lists B.R. Fisher at , with the same telephone number as Looky Video. The 1982/3 Amateur Radio Callbook lists a B.R. Fisher, VK3YRF, at . Old ads in 'E.A.' list Bruce Fisher, VK3YRF, as the principal of Looky Video.

I didn't draw this together last July, but am now considering the possibility that my customer Bruce Raymond, and Bruce R. Fisher, both of 2 Herbert St., Watsonia, inhabit the one body. If so, then I must conclude that one Bruce purchased the chargens with the intention of advertising and selling rip-off clones after learning "through KAOS Club" of their availability.

The same Bruce Raymond also wrote to me on 2.11.82 for details of the "2 VIA Eprom Burner" after I wrote about it in KAOS. Looky Video sells "Collected Burner Circuits" for $9.95. While I have not seen these notes, I have an open mind on what might be supplied. Also advertised are "Add 8k Piggyback RAM" notes for $1.95. These would be sound value if they rely on the same principle described in "Add 8k Piggyback RAM" in OSUG newsletters 14 and 24.

In case you don't know how to organise your own sales, here are a few hints from the catalogue.

"..There are a lot of good ideas out there so let's hear about them!...We pay commissions for any marketable items. Referr to our page on submitting items."

Sadly, the catalogue contained no such directions. Perhaps the page had been borrowed.

CHARGEN PRICES FROM 1.1.84

To encourage the Qld. economy, revised chargen prices are:
2716 $10.50 ; 2732 $14.00 . For orders of > one chip, discount 10%.
Prices are inclusive of postage and packaging.
B. Wills,

— SUPERBOARD —

SOFTWARE REVIEW - Monster Maze

Monster Maze is a 4½k M/C arcade game occupying $ØDD2 to $1FFF. The game is very similar to Atari's Pacman, and from the sales point of view, Pacman arcade games have been high on the list of money earners. Pacman has been regularly advertised on television, and for the above reasons, is popular with arcade owners.
There is a two dimensional maze on the screen. Along the pathways of this maze are a regular pattern of dots. In each corner there is a diamond shaped lozenge. Patrolling these paths are four creatures which look rather like jet planes. (OSI graphics does have it's limitations)

The player uses four keys to control the direction of movement of a small man. To score, you move him along the pathways, "eating" the dots, and avoiding the jets, which attempt to find and kill him. To assist your evasion, there are two exits, left and right of mid-screen. If you enter one, you re-appear on the opposite side. A good way to score extra points is to eat a lozenge. Magically, the jets turn into castles (CHR$(4)), and the chase is reversed. The first castle you eat is worth 200 points, the next 400, and so on. However after a few seconds, and without warning, the castles revert to jets and are after you again. Should you be caught, there is a delay (moment of triumph for the computer), then you get another man. When three men have been lost in this manner, the game ends, and you read off your score. To play again, you hit the "R" key.

There seems no limit to the number of screenfuls of dots and lozenges that you can eat, provided that you are skilful enough to avoid the jets.

The keys used are ESC=UP, CTRL=DOWN, R/SHIFT=RIGHT, /=LEFT. While a little difficult at first, you soon master these keys for movement when there is no pressure on. However when several jets are hot on your tail, the brain seems to switch to subconscious mode, and your fingers behave differently, this often resulting in your man being gobbled in double quick time.

The game can be converted for joysticks with a few simple changes to the code. I have heard that Looky Video still sells Monster Maze. Perhaps if they would be kind enough to send me a catalogue, some of these reviews might sell them a few games? Address is P.O. Box 254, Heidelberg, 3084.

The OSUG library has a 24 x 24 version at the usual postage rates.

- -

NEAT EXTRAS

Bernie Wills has written some Basic Extras which locate at address $9800 in an eprom. The extras include Screen Fill, Printat, Usr Call, Horizontal and vertical Plot, and Block Fill. Originally he implemented the routines by using the modified parser technique which was pioneered by Ed Carlson, (Hooks into OSI Basic). They can be used in program or immediate mode.

Now, with a modification to the Basic 1, he has incorporated his extras into the language, so that the modified parser is not needed. NULL is a little used function, and can easily be executed with a POKE to location 13 (decimal). The word NULL is located at $A0C5, and is replaced by:-

```
        $A0C5  42 57 45 BD
```

The actual NULL code starts at $A67B, and this too is changed:-

```
        $A67B  4C 00 98 EA EA - - - continue EA to $A68B inclusive.
```

Now, by using single key basic, you get BWE= instead of NULL. Instructions become BWE=S187 to fill screen with CHR$(187), BWE=U$FD00 to call getkey, and similarly for the other functions.

Ed Richardson.

ALPHA 80 NOTES

by Frank Halley

I purchased my Alpha-80 a few months ago from George, and have it hooked up to my pseudo C4 as device #4 (parallel printer port). I have been very impressed with the capabilities of the printer, and can recommend it, especially at the price George was asking recently!

I left the DIP switches the same as the factory settings shown in the manual, except for SW1-7 ON (slashes zeros) and SW1-3 ON (enables one inch skip over perforations).

Like Frank Brown (KAOS 3/11) I found the screen dump in OS65D V3.3 didn't work properly, and set out to investigate whether it could be fixed to work properly with the Alpha-80. The short BASIC program below pokes in some changes, and the screen dump command ("PRINT!(80)") now works properly for text on the screen. A full screen dump including all the OSI graphics set would be possible with the printer's bit- image printing but would take a bit of writing! Before giving the screen dump command, the printer must be activated (eg. 'DISK!"IO ,08"') otherwise you will get a screen dump to the screen - not particularly useful!

One thing I didn't like about OS65D V3.3 was the way it kept track of the page length for the printer. The printer also keeps track of this, and I got some funny effects near the bottom of a page - the computer would send a few line-feeds to get to the top of a new page, but the printer would also skip one inch over the perforations! This caused my new page to begin well below the top. The BASIC program contains a couple of pokes to fix this too - it just disables the part where the computer counts how many lines have been printed, leaving it up to the printer. Having the one inch perforation skip set means I have to tell WP6502 that my pages are actually shorter than they really are, so it just feeds to near the bottom of the page and the printer takes over.

The last poke is so the parallel port can be enabled by CONTROL-L (a fact not mentioned with my Compdos 1.3 disk). This was set to enable the serial port (device #1) as sent from Compsoft.

One last note - OSI's use of the ASCII code doesn't quite agree with everybody else! OSI use character 124 as the right curly bracket and 125 as a vertical broken bar - everyone else (including the Alpha-80 printer) uses these codes the other way around.

```
10 REM PATCHES TO FIX OS65D V3.3 SCREEN DUMP PROGRAM
20 REM TO WORK PROPERLY WITH ALPHA-80 PRINTER
30 REM
40 POKE 12900,10 : REM TO FORCE L/F AT END OF EACH LINE
50 POKE12761,17  : REM CHARACTER FOR BORDER
60 POKE12759,17
70 POKE12905,17
80 POKE12839,17
85 POKE12841,17
90 POKE12841,17
95 POKE12888,0
100 REM
110 REM
120 POKE12801,208
130 POKE12802,15
140 FORX=12781TO12790:POKEX,234:NEXT : REM SOME "NOP"S
150 FORX=12877TO12879:POKEX,234:NEXT
160 REM
170 REM NOW FIX LINE FEED PROBLEM !
180 REM
190 POKE9670,0
200 POKE9684,226
210 REM
220 POKE9618,8 : REM ENABLE CTR-L TOGGLE FOR PARALLEL PORT
225 REM
230 REM NOW DO SCREEN DUMP
240 DISK!"IO ,08" : PRINT!(80) : DISK!"IO ,02"
```

CMOS BATTERY CLOCK FOR THE 16 PIN I/O BUS

by David Tasker

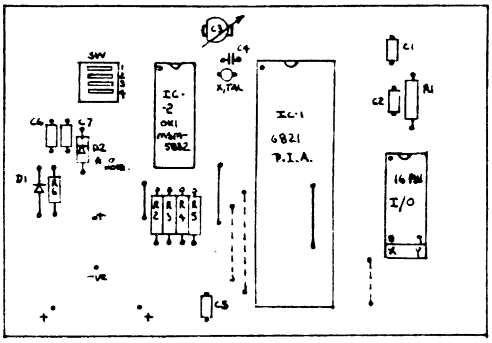
Last month we discussed the inner-workings of the 16 pin I/O bus. This month I have provided the circuit of a CMOS clock using the OKI MSM5832 4 bit RTC (Real Time Clock).

You may have heard that OSI supports a real time clock. In fact the OSI clock is quite simply an interrupt signal generated at regular intervals which interrupts the CPU chip from its current job and forces it to JSR (jump to subroutine) at $01C0 where it expects to find a program that will process these regular interrupts and keep the time. This system works fine except for one major limitation. Each time your computer is switched off or "hangs up" the time is lost, the program must be re-entered and the time reset.

The OKI MSM5832 chip is a CMOS device that can be kept powered independant of the host computer by a small battery. Internally the Clock chip has a number of registers that have the current time, date, year and also AM/PM, 12 or 24 hour mode and leap year information. It is a fairly simple matter to write software in BASIC to program the time into the clock and later read it out and convert the information into day and month information.

Because the 5832 chip is a little slow it is not easy just to tie the chip directly onto the CPU bus. Also we need to generate a couple of control signals for a period of time which is longer than our 1MHz or 2MHz CPU clock. Incidently the clock has its own crystal running at 32768Hz which if divided by 2 -15 times happens to give 1 pulse every second. How lucky can you get?

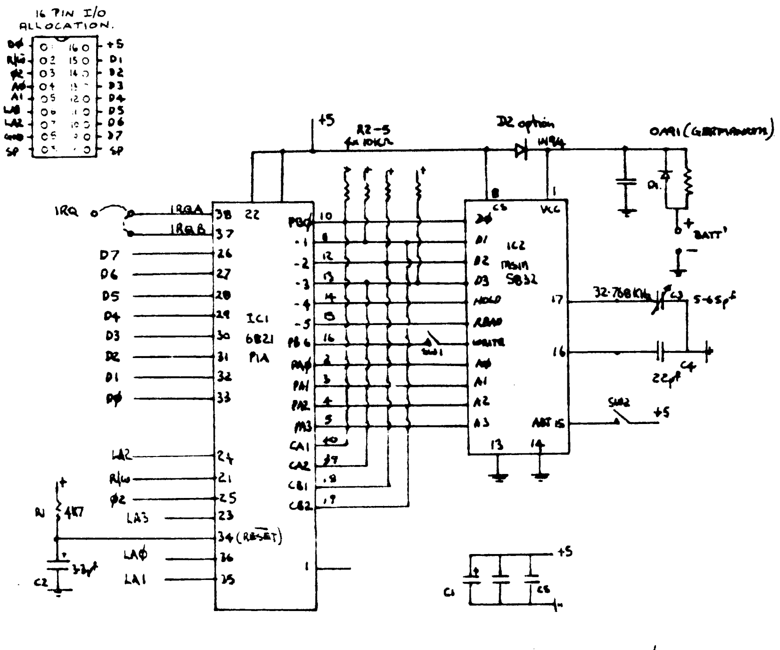
The circuit consists of a 6821 PIA chip to interface the MSM5832 to the CPU. The DIP switch is used to "Write Protect" the clock. A Battery which can be either 3 volt lithium or 3.6 volt Ni-cad. A 32.768KHz Crystal and a few resistors and capacitors.



**** PARTS LIST ****

IC 1. PIA 6821
IC 2. MSM 5832 (OKI)
R1 4.7K Resistor.
R2 100 ohm Resistor.
R3-R6 10K Resistors (4).
D1 OA91 Diode.
D2 IN194 Diode.
C1,C2 3.3uf Capacitors.
C3 22pf Capacitor.
C4 5-65pf Adjustable.
C5,C6,C7 0.47uf Ceramic.
32.768KHz CRYSTAL.

Battery may be 3.6volt Ni-Cad (insert D2).
or CR2032 3Volt Lithium. D2 Must NOT be in.



16 PIN I/O
REAL TIME CLOCK.

```
10 REM ===========  TIME*  ===============
45 GOSUB10000
52 PRINT"This program is used to set an MSM5832 Clock
54 PRINT"I.C. ":PRINT:PRINT"It assumes a PIA at $C600"
56 PPINT"Time and Date information will only be input"
58 PRINT"to the MSM5832 if DIP Switch 1 is on. This"
60 PRINT"Switch is located on the Clock I/O Board."
70 PRINT:PRINT"Enter the Time and Date as shown in the prompts"
80 PRINT"Corrections are allowed after entry of all Data"
90 PRINT:INPUT"DO YOU WISH TO SET THE TIME";ST$
95 PRINT:PRINT:PRINT"PORT AT 1) $C600"
96 INPUT"          2) $C704";PO
97 IFPO=1THEN101
100 PA=50688+260:PB=50690+260:GOTO110
101 PA=50688:PB=50690
110 REM RESET PORT
120 POKEPA,0:POKEPA+1,0:POKEPB,0:POKEPB+1,0
121 IFLEFT$(ST$,1)="N"THEN3000
122 POKEPA+1,255:POKEPA+1,4
123 POKEPB+1,240:POKEPB+1,4
150 GOSUB10000
151 PRINT:PRINT"Answer questions with Numbers unless prompted"
152 PRINT"otherwise. e.g. Months = 07 (July)"
154 LY=0
155 INPUT"IS THIS A LEAP YEAR";LY$:IFLEFT$(LY$,1)="Y"THENLY=1
160 INPUT"YEAR ";Y$
161 INPUT"MONTHS";MTHS$
162 INPUT"DATE ";DATE$
163 INPUT"DAY OF WEEK";D$:D=VAL(D$):GOTO174
164 INPUT"Do you want 12 or 24 hour mode (12 or24)";MODE
165 PRINT:IFMODE=12THEN400:MODE=24
166 INPUT"HOURS";H$
167 INPUT"MINUTES";MINS$
168 PRINT"Seconds will be reqested as 00 when it is time to"
170 PRINT"start the clock"
171 FORX=1TO1500:NEXT
172 GOTO300
174 IFD<1ORD>7THEN162
175 OND GOTO201,202,203,204,205,206,207
201 D$="Sunday ":GOTO164
202 D$="Monday ":GOTO164
203 D$="Tuesday ":GOTO164
204 D$="Wednesday ":GOTO164
205 D$="Thursday ":GOTO164
206 D$="Friday ":GOTO164
207 D$="Saturday ":GOTO164
300 GOSUB10000
302 RESTORE '
305 FORX=1TOVAL(MTHS$):READM$:NEXTX
310 PRINT
311 Y=VAL(Y$)
312 PRINT"THE DATE IS...."; D$;M$" ";DATE$" 19";Y
313 H=VAL(H$):MINS=VAL(MINS$)
320 PRINT:PRINT"THE TIME IS ... ";H;":";MINS;": 00";
322 IFMODE=12THENPRINTAMPM$
330 PRINT:INPUT"Is this information correct";I$
340 IFLEFT$(I$,1)<>"Y"THEN160
350 GOSUB4000
360 INPUT"ENTER ANY KEY WHEN YOUR WISH TO START THE TIME COUNT";TC$
370 POKEPA,0:POKEPB,80
380 POKEPA,1:POKEPB,80
390 POKEPB,0
395 GOSUB3000
400 INPUT"AM or PM";AMPM$
401 GOTO166
2000 DATA"January","Febuary","March","April","May","June"
2010 DATA"July","August","September","October","November"
2020 DATA"December"
3000 REM READ TIME
3020 REM SET UP LS4 DATA LINES AS INPUTS ON PIB
3030 POKEPB+1,0:POKEPB,240:POKEPB+1,4
3040 POKEPA+1,0:POKEPA,255:POKEPA+1,4
3050 POKEPB,48
3060 POKEPA,0:SU=PEEK(PB)-48
3070 POKEPA,1:ST=PEEK(PB)-48
3080 POKEPA,2:MU=PEEK(PB)-48
3090 POKEPA,3:MT=PEEK(PB)-48
3095 POKEPA,4:HU=PEEK(PB)-48
3097 POKEPA,5:HT=PEEK(PB)-48
3099 POKEPA,6:W=PEEK(PB)-48
3101 POKEPA,7:DU=PEEK(PB)-48
3103 POKEPA,8:DT=PEEK(PB)-48
3105 POKEPA,9:M2=PEEK(PB)-48
3107 POKEPA,10:M1=PEEK(PB)-48
3109 POKEPA,11:YU=PEEK(PB)-48
3111 POKEPA,12:YT=PEEK(PB)-48
3200 POKEPB,0
3500 RESTORE
3505 AMPM$="AM"
3510 W=W+1:ONWGOTO3900,3910,3920,3930,3940,3950,3960
3520 GOSUB10000
3522 GOSUB3800
3530 PRINT"      CURRENT TIME IS    ";HT;HU":";MT;MU":"ST;SU" ";AMPM$
3540 PRINT
3550 M3=(M1*10)+M2
3555 FORX=1TOM3:READM$:NEXTX
3560 PRINTTAB(5);W$;M$"";DT;DU" 19";YT;YU
3590 FORX=1TO14:PRINT:NEXT
3592 INPUT"ENTER ANY KEY TO READ TIME AGAIN";AK$
3595 IFAK$="TIME*"THENRUN
3596 IFAK$="ABORT"THENRUN"BEXEC*"
3600 GOTO3020
3800 IFHT<4THENGOTO3850
3805 IFHT=5THENGOTO3855
3810 IFHT=14THEN3860
3815 IFHT=13THEN3870
3820 IFHT=9THEN3870
3830 IFHT=4THEN3880
3850 AMPM$="AM":RETURN
3855 HT=1:AMPM$="PM":RETURN
3860 AMPM$="":HT=2:RETURN
3870 HT=1:AMPM$="":RETURN
3880 HT=0:AMPM$="PM":RETURN
3900 W$=" Sunday ":GOTO3520
3910 W$=" Monday ":GOTO3520
3920 W$=" Tuesday ":GOTO3520
3930 W$=" Wednesday ":GOTO3520
3940 W$=" Thursday ":GOTO3520
3950 W$=" Friday ":GOTO3520
3960 W$=" Saturday ":GOTO3520
4000 REM WRITE TIME
4010 POKEPA+1,0:POKEPA,255:POKEPA+1,4
4040 POKEPB+1,0:POKEPB,255:POKEPB+1,4
4080 YT=VAL(LEFT$(Y$,1)):YU=VAL(RIGHT$(Y$,1))
4090 POKEPA,12:POKEPB,80:POKEPB,80+YT:POKEPB,YT
4095 POKEPA,11:POKEPB,80:POKEPB,80+YU:POKEPB,YU
4100 M1=VAL(LEFT$(MTHS$,1)):M2=VAL(RIGHT$(MTHS$,1))
4110 POKEPA,10:POKEPB,80:POKEPB,80+M1:POKEPB,M1
4120 POKEPA,9 :POKEPB,80:POKEPB,80+M2:POKEPB,M2
4150 DT=VAL(LEFT$(DATE$,1))+LY:DU=VAL(RIGHT$(DATE$,1))
4160 POKEPA,8 :POKEPB,80:POKEPB,80+DT:POKEPB,DT
4170 POKEPA,7 :POKEPB,80:POKEPB,80+DU:POKEPB,DU
4200 IFMODE=24THENMODE=8
4202 IFMODE=12THENMODE=0
4208 IFAMPM$="PM"THENMODE=MODE+4
4210 HT=VAL(LEFT$(H$,1)):HU=VAL(RIGHT$(H$,1))
4220 POKEPA,6 :POKEPB,80:POKEPB,80+(D-1):POKEPB,D-1
4230 POKEPA,5 :POKEPB,80:POKEPB,8Q+HT+MODE:POKEPB,HT+MODE
4240 POKEPA,4 :POKEPB,80:POKEPB,80+HU:POKEPB,HU
4300 MT=VAL(LEFT$(MINS$,1)):MU=VAL(RIGHT$(MINS$,1))
4310 POKEPA,3 :POKEPB,80:POKEPB,80+MT:POKEPB,MT
4320 POKEPA,2 :POKEPB,80:POKEPB,80+MU:POKEPB,MU
4500 RETURN
10000 FORCL=1TO32:PRINT:NEXT:RETURN
63999 DISK!"PU TIME*"
```

MDMS REVISITED
by Norman Bate

Two problems have been encountered when using OS-MDMS for tasks requiring larger than average amounts of data. Each problem is discussed and a solution given. The task that revealed the problems was to set up a competitor file for a Fun Run. Approximately 10,000 records were to be stored at the rate of 800+ per disk (5.25") on tracks 13-39 leaving tracks 01-11 for keyfiles.

The small problem required the writing of another auxiliary program to fix, and is discussed later.

The large problem occurred when doing a keyfile sort. The program appeared to do its stuff ie. reading, sorting and rewriting. However, when a keyfile dump was done, the rewritten data was still in its unsorted form.

After much frustration the full extent of the problem revealed itself after a complete keyfile dump was printed out. The keyfile occupied 4 tracks (01-04). Sorted data started to appear towards the end of the printout. Use of the EXAM function showed that in fact it started on track 04.

What was happening was that sorted data was being rewritten to 4 tracks alright, but using the 'current track' in OS-DOS as the start track and so was rewriting to tracks 04-07. The cure was to add the following lines to the KEYFILE SORT program.

```
        1055 disk CLOSE,6
        2055 DISK OPEN,6,FI$
```

This will reset the 'current track' in DOS to the start of the file FI$. If another line: 2045 PRINT I;:PRINTCHR$(13);
is added, an incremental count as the sort advances will be seen in the BLH corner. Delays of several seconds will occur as garbage collection is performed.

If you use a printer then it is most annoying to have a line of dashes (-) across the page after each 22 lines of printout. These dashes can be eliminated in both the REPORT WRITERS by changing one line in each as follows

```
        381 PT=PT+1:IF P<60 GOTO 400
```

My Microline printer has a reduced print capability so I have modified both of the REPORT WRITERS to incorporate this facility. When page width is asked for in line 280 I check for an answer>80. If this is detected the printer is put into the reduced print mode and restored as follows

```
        282 IF DW>= 80 THEN PRINT #1,CHR$(29)
        1050 NEXTI:PRINT#1,CHR%(12);:PRINT#1,CHR$(30);:GOTO4000
```

The small problem that I referred to earlier occurred because all of the Master file must be used. Having selected an alphabetical grouping of say A-D and after typing some 700 records in random order it became obvious that all the records would not fit in the file. With the current set of programs the only way out was to retype a new alphabetical listing using only A-C.

I have written a program that will create a new Master File using some or all of the records in a current Master File. You first CREATE a new Master File (using the same name as old file) and then run the new program PARTIAL FILE TRANSFER. You are asked for the record numbers to start and end the transfer and away it goes. In the case of the FUN RUN problem the old Master File was sorted and then the appropriate record numbers used for the tranfer.

I have written another program that will allow a Master File to have one field in all records updated by a +/- number or percentage. This is very handy if you have inventory type data in the file and want to increase a price field by 5% or increase the min. stock by 1. Of course EDITOR can still be used for changing the field of a single record.

Listings of the programs will be available from the library or by contacting me direct, KAOS has my address.

Brisbane User Group Meeting, 15th Jan. 1984

Attendance: 17   Computers: 6 (all working)

It turned out to be an excellent day for an indoor meeting. The weather was just awful, with light to heavy rainfall all afternoon.

The meeting proper started at 1pm. I had planned for something a little unusual this time, and arranged for some different types of machines to be there. A Microbee and an Apple 2E failed to turn up, unfortunately.

John Froggatt was the first to arrive, carrying his C1/4 under one arm, and his new BBC/B under the other. John had built a neat timber bridge over the case of the BBC to hold his monitor. He had an excellent assortment of graphics display and arcade game programs, and the sound from the latter demanded attention. Ross Beneke was particularly enthusiastic, and seemed the only member capable of mastering some of the more complex games.

Brendan Vowles gave an excellent demonstration of the capabilities of, and facilities offered by The Australian Beginning. The lack of any software for OSI was a bit of a drawback. The downloading program seems to have stalled. Members offered some suggestions for the joke file which met with much local approval, but were deemed unsuitable for a public utility. Finally, one was chosen and inserted with much help from TAB.

Alan Calvert had just moved house, and has had little time in which to play computers of late. One program he had been working on played a couple of lively tunes on the soundport, which attracted much interest.

Harry Moores had his new BBC/B with disk drive, and was more interested in the serious side of programming, and also the better graphics capability of the BBC. Harry had a colour T.V. converted to R.G.B. operation, and the resolution, even on 80 characters width, was excellent. He expressed regret about parting with his Superboard, but felt that sentiment had to give way to progress. Hopefully, someone else could use it to derive as much pleasure and knowledge as he had done.

Bernie Wills had his C1/4, and gave a demonstration of his Brother CE60 correctable daisy wheel typewriter and serial interface unit. The machine gave IBM quality print and seemed value at $650. When driven by the computer control characters can be sent to drive the carriage in any direction, plus various other functions.

The meeting finally ended around 6pm.

KAOS-WA

by Gerry Ligtermoet

I do not have much to report from our January meeting. It was attended by fewer members than normal due no doubt to the holiday season.

However Peter Hughes did indicate that he had finally got his 8K RAM board up and running. Some members indicated that they were interested in running parallel printers and were wondering how to do it. I can now report that I have managed to get mine running with an interface board from COMP-SOFT using the ACIA on the CPU board.

Our next meeting is on Sunday March 18th at 2.00pm at Guild House, 56 Kishorn Rd, Mt Pleasant. For more information contact me at

T.A.B. NEWS

Dear KAOS members,

Welcome to 1984. It may be George Orwell's year of significance, but we think that it's also ours.

Since the last newsletter, substantial changes have taken place at the Australian Beginning - for example, the long awaited Data General MV 8000 has been installed and Version 1 of the connection to Austpac is available to all members throughout Australia. No longer do interstate members have to pay S.T.D. rates when they access the system.

For your information, the MV 8000 hardware configuration is as follows:- Two (2) M/Bytes of Main Memory, 400 M/Bytes of rotating Memory (another 400 M/Bytes on order), 64 Communication Ports (Direct Dial-Up), four (4) Austpac Interface Ports allowing 16 users per port and many other goodies. The operating system is the latest version of AOS/VS.

Other improvements to the User work area were announced at the last User Group meeting. These were:

A new version of Chat-Mode - more powerful and easier to use. (An Andrew Lighten Masterpiece).

A new Text Editor - much improved over the old version (another Andrew Lighten Masterpiece).

These User Groups' Meetings are held on the third Wednesday of each month at 24 Camberwell Rd, Hawthorn East at 7p.m. - so try to attend. The more people who attend, the greater the value of discussion.

Best Regards to All,
Noel S. Fenton.
Chief Executive Officer.
The Australian Beginning Pty. Ltd.

THE MEETING WAS KAOS

by King Corky

Well, here we are at the start of another year. Let's hope this one is a bit better than the last, (no more Ash Wednesdays please).

This month's meeting, the first for 1984, went rather smoothly with some hardware for sale..viz..a load of vector graphics systems, (ex video games machines), were up for grabls at $75 each including a 20" screen and a couple of Asteroids boards. By now, most should be sold but check with David Anear. There were also a few other video games boards for sale, see under FOR SALE. Paul Dodd from COMPSOFT, (*!%$*%!), almost has the CP/M board for the RABBLE, (& OSI), ready for sale. Final testing has to be done and the lot, including the new floppy disk controller, (for both mini & 8" concurrently) should go for around $295. The double density floppy controller, (for CP/M, Flex, Compdos 1.3), is ready but at present there is no driving software. Pull the fingers out boys, I'm still waiting.

Better Electronics has the 65C02 chip available and Ray Gardiner would like a download routine for Apple to OSI, (would like to read Apple disks with the Rabble/OSI), (wouldn't we all).

A little tip for those of us who lump our precious, precarious, finicky, temperamental, downright obstinate, systems to the meetings with the hope of demonstrating some new hardware/software we have incorporated, or the hope of finding something new we could add to our own supplies. If you ever have trouble with the BRUTE when you get it home again then I suggest you make/buy yourself a card-cage that will hold everything tightly in place and once installed, NEVER, repeat NEVER, remove it, (unless you are making more mods).

I had made some mods to my system over the Xmas holidays, (added an RGB driver board), and didn't bother to refit the top of the card-cage before zipping off to the Jan. meeting. Once setup at the meeting, everything appeared to be functioning OK, but when I got the Monster home, I discovered I had no signals to drive my printer PIA. There was no 02 clock pulses to strobe the PIA, no data, no nothing, DEAD. After hours of bug chasing, pulling-out & re-seating boards, power supply checks, logic probe poking, (which also created a bug or 2), etc. etc., I finished up with nothing. Absolutely nothing. No system on-line at all. DOUBLE DEAD.

The first thing obviously was to get SOMETHING happening. Power supply, loose connection, OK, 02??, who knows??, pull out the RAM/PIA board AGAIN, get out the magnifying glass, look, and look again, and again, EUREKA, a broken joint on one pin of a piggy-back arrangement, (installed by someone else), a fault very hard to find, but one with disastrous reults. All caused by the board flapping around on the bottom-half of the card-cage and banging against the board next to it. NEVER AGAIN will I move the Monster without first bolting everything down and then completely covering the whole lot (stock-and-barrel) with epoxy resin, PVC tape and then cement, (just to make sure).

OSI HISTORY Part 3

by Eric Lindsay

Like most disc operating systems, OS65D3 contains routines for putting Basic (and other language) programs to disc, and loading them back into the working memory. Since it is intended as a program development oriented DOS, rather than a business users' DOS, it also contains routines to save particular areas of memory (which would normally contain machine code routines) onto specified areas of a disc. Naturally it also allows you to call such areas back from any track and sector of the disk and place the contents in any area of memory.

You also have some fairly direct control of the disc, with commands such as HOME, which sets the disc drive head to track 0. Since you can insert discs while the disc motor is running, it is fairly easy to move the read/write head, thus confusing the DOS as to where the head is positioned - HOME provides a convenient way to deal with this situation. You have the option of loading various languages, such as BASIC, ASM (the 6502 Editor Assembler), or EM (the Extended Monitor and disassembler). You can EXIT directly to the operating system command level, and return to your undisturbed code by using RET BAS, RET EM, or RET ASM. You can also use any operating system command from within BASIC by prefacing it with DISK!"command string", or from within the Extended Monitor by prefacing it with !

Preparing a new disc is done by IN, however if you have problems with reading a disc, you can initialize a specific track using IN nn, where nn is the track number (in decimal). You can also use the DOS command GO nnnn (nnnn is a hexadecimal number) to execute the code starting at nnnn. Since this can be done from any language, this takes the place of the machine language CALL statement that is available in Applesoft.

The facilities for the machine code programmer are rather nice, since you can CALL any track and sector and place it in any area of memory. Using the Extended Monitor you can then disassemble it, change it, and then SAVE it back to disk. If you are having problems, or are simply curious, you can use the EXAM nnnn=TT command, which take the entire contents of track TT (decimal), and places it into memory starting at nnnn (hexadecimal - I didn't say the commands were terribly consistent). Dumping the track into video memory, for example, is thus made very simple. You can easily incorporate the command in a simple BASIC program, so you can inspect the total contents of a disc in a few minutes. This is handy when you forget what you have on each track, or accidently destroy the directory. The EXAM command absolutely ignores all safeguards, and displays everything on the track, including formatting information. You can even read blank, uninitialized discs with it. You won't get much sense from them, but it will try to read them nonetheless.

A few other features are worth mentioning, as background. You can change the input/output distributor flags with IO nn,mm. This means that you can send video output to a printer, cassette, disc, memory, or a serial terminal. You can also take input from the keyboard, serial terminal, a cassette, disk, or memory. You get some strange effects at times if you try taking input from more than one device at once, but even this can be done. This means that you can easily imitate some fancy features of more expensive operating systems. Send your typed commands to memory from the keyboard, and then arrange them to be input from memory during a program, and you have the equivalent of a CP/M Submit file. Or you can send output to video, printer, cassette and memory all from the same program run, thus providing some of the pipelining features of Unix.

Obviously there have to be some disadvantages, and they become transparently obvious to even a rank beginner when you first try to PUT a BASIC program onto disc. The operating system accepts six letter file names, and will LOAD and RUN named files, however it does not let you PUT a named file to disk until the file already exists on the disc. OSI suggest that you use their CREATE utility (a program written in BASIC) to make a suitable file heading, such as TEMP, and save your work in this TEMP file. The big problem is, if you forget to create the TEMP file, you will destroy your program when you LOAD the CREATE utility into the working memory. OSI's operating system lets you PUT files by track number alone, but then they don't appear in the directory, and you soon lose track of where they are.

There are also a bunch of other utilities, also written in BASIC, that accompany the OS65D3 operating system. These include DELETE, which deletes a file name from the directory. RENAME, whose purpose is obvious, and a whole set of DIRectory utilities. These are BASIC programs which list the directory in the order in which it was created, or list the directory in alphabetical order by filename, or by track number, rather than in the order in which files are created.

Not having all that good a memory, I was always forgetting to create files, or forgetting which track I had saved them on, or deleting the wrong file, or in some manner destroying programs I was trying to save. The solution to this is to have the computer do all that work, the way it does in AppleDos, TRSDOS, or CP/M. The program that adds the needed facilities is called Comp-Dos, and it is sold by Compsoft, at 235 Swan Street, Richmond Victoria (I have to mention the name, because the people there wrote it, and I don't know where else it can be obtained).

There are a few different versions of Comp-Dos. Some time before it was produced, David Anear of the Victorian OSI user group, KAOS, wrote a very nice full screen editor for OSI systems. This was sold in EPROM as a replacement for the OSI monitor ROM, since the editing facilities in the original monitor were almost non-existent. This screen editor is included in Comp-Dos 1.2.

Continued next month

FOR SALE

Teletype ASR33, ASCII printer/terminal, 110 baud, $150 ono. Located in Melbourne. Vic Mc Donald,

Superboard Series II, Dabug 3, 8K RAM, Tasan video board, 1/2 MHz, 300/600/1200 baud, power supply, manuals, various software (inc. Assembler, Ext Mon, Adventures, games etc.) $450 ono David Halsall

Superboard with 16K RAM, Cegmon, RS232 and built into a home made metal box complete with power supply, (the keyboard is not covered). All documentation and some programs inc. Has been used mainly as a RTTY terminal. $150 ono

One C2 OEM back plane with 8 x 48 pin OSI bus sockets, 3 boards plugged in. They are: 505 Processor board, 535 RAM board with 48K, 550 interface/expansion board. A Power-One HCAA-60W power supply is hard-wired to the back plane. Was a going system when removed from service. Was interfaced with 2 Shughart 801 drives and a serial printer/terminal. Documentation included. Best offer gets it. Gordon Dowse

OSI Superboard 2 series 2, dabug 3, Rabble Expansion board (40K RAM, 32K ROM, Floppy disk controller, dual sound drivers), cassette interface with multiple baud rate, switchable to 1-2 MHz. Assembler Eprom, comes with manuals and software. $500 ono Andrew Wale

48 pin, 24K RAM board, populated to 16K, (with 2114s), fully operational, $100.

Ex Telecom direct-connect modem, you can have it for what I paid, $75.

240V/240V mains filter/regulator, $70.

Hitachi RGB to composite color, (NTSC, Yanky color), converter, two outputs, compete in a very nice case, at cost $50.

Ex Queensland, (ex Superbord User Group), direct connect, non-standard frequency type, modem at cost, (mailing charges only), $5. Both modems come with full set-up details and circuits.

Model 15 teletype with interface but no power supply, has a minor bug in the ever-so-simple interface, $50.

WANTED: EHT transformer for Philips color TV, K9 chassis, (and possibly someone to help with installation).
See Ron Cork at sale next meeting or ring

SUPERBOARD SERIES 1 with case and power supply, 16K RAM, Dabug 3, Joystick, Tasker Bus includes 8K RAM card, 3-AY-8910 sound board with software driver, PIA/VIA card, Eprom card with 10 ROMs, 16 pin expansion card, Eprom burner (2708-2764). 30+ cassettes of software, extensive documentation. $450 ono.

STAR dot matrix printer, 80 columns, various type densities, centronics interface. $350

Philips R.G.B. and composite video 14" colour monitor (no case) $300 ono

10 video game boards, Amigo, Scramble, Galaxians, Pacman etc in working order, can be used with a B/W monitor or can be used for parts. A QIX board has 2x6809, 6x6821, 12x2532, 1x6845 CRTC, 64K 4116 RAM, and about 100 TTL and CMOS chips. $75 each ono Ian Forrest

Registered by Australia Post
Publication No. VBG4212

If undeliverable return to
KAOS, 10 Forbes St
Essendon, Victoria 3040